IMPLEMENTING ACQUISITION REFORM IN SOFTWARE ACQUISITION

1. Reduce Cost Of Ownership

Any system that is developed goes through the same standard development phases: system concept, feasibility studies, development, production, and support. Software, as part of the system, is included in each of these phases except for possibly in system concept. In most cases where software development is involved, new software programs, or Computer Software Configuration Items (CSCIs) will be developed. Software development costs in modern systems can be substantial, both in relation to the hardware and in absolute terms. In recent years, hardware components have been reduced in size and cost, and in some cases even become 'standardized'; whereas software in modern systems is very large and complex, and frequently takes longer and costs more than the hardware to develop. When considering the costs, both the development costs and support costs must be taken into account.

In the past, software has been developed on an 'ad hoc' basis, with few well-defined development processes used. This led to software that is delivered late (and sometimes not delivered at all), software development that costs twice as much as expected, and software that does not meet the stated requirements. Most problems associated with software development and support can be directly traced back to deficiencies in the way that the software was planned, managed, and designed. In order to reduce the costs of developing software, there are four major things that need to be done:

- Define the software development process. If the software development process is not defined, the software may be developed in a chaotic, ad hoc manner with concomitant loss of cost control. Development managers may not be able to trace costs, schedules may not be met (if they can be determined at all), and management may have no idea of what is really going on. A well defined software development process allows management to track the progress of the effort and take corrective action if something unexpected happens. Also, it allows for costs to be tracked, analyzed, and controlled.

- Have requirements traceability. Most of the problems with software development relate to the requirements: the requirements are not implemented correctly and the system does not perform as intended, some requirements are not implemented at all, some functions that may be essential are not foreseen and therefore are not included in the specifications for the system, and some requirements may be added to the 'needs list' after the system is well under development. Each requirement, whether operational requirement, user requirement, or support requirement, must be traceable from the Operational Requirements Document to the specifications to the design to the code. The addition of 'forgotten' requirements and changes to implemented requirements that do not perform correctly are costly and time consuming.

- Develop useable documentation, for the users as well as for the maintainers, in parallel with (actually as a by-product of) the software development. After the system is developed, sufficient documentation is needed to employ the system efficiently with a minimum of operator labor and expense. More importantly, support documentation must also be developed. After the system is deployed, it will be supported for its entire life cycle, which can be 20 years or longer. This documentation must be understandable by individuals who did not develop the system so that they can efficiently support it during it's life-cycle. .

- Ensure that the software is designed to accommodate change. Any system that is deployed will undergo change during its life, e.g., correction of errors, addition of new requirements, updating the technology in the system. If the developed software can not accommodate change, then it will be difficult or impossible to continue supporting the software throughout the life of the system. This can mean that the software will need to be developed again at great expense and/or delay, or that necessary changes will not be made.

Once the computer resources in the system , including the software, have been developed and deployed, they must be supported throughout the entire life of the system. This involves primarily the support of the software in the system. Studies have shown that the typical cost to maintain software is from 60% to 80% of the total system life cycle costs. This support includes incorporating new requirements into the system as well as correcting errors, developing software for ECPs to the system, testing the changes, integrating the changes with the rest of the system, recompiling the updated software, delivering the software to the user, and possibly installing the software at the user site. To meet these support requirements, appropriate statements need to be included in the RFP which allow for the computer resources in the system to be supported by Government activities.

Until recently, DoD required software developers to obtain a waiver to use any programming language other than Ada 95 in new software. The current policy is that Ada must be considered when selecting a programming language, but the use of another language does not require a waiver. Ada was developed to provide a common programming language that would reduce development time and support costs during the life cycle of the system. If Ada 95 is selected as the programming language, then statements must be included in the SOW requiring the contractor to write the software in Ada 95, as defined by ANSI/ISO/IEC-8652:1995 "Information technology -- Programming languages -- Ada." The system specification(s) must also state that the software will be developed in Ada and that this will be one of the acceptance requirements. ANSI/ISO/IEC-8652:1995 replaced ANSI/MIL-STD-1815A which defined the Ada 83 language.

Increasingly, software that is used in the system, or to support the system, may be Commercial Off-The-Shelf (COTS) software. When this software is purchased, technically, the purchaser is buying only a license to use the software, not buying the software itself. This license usually limits the scope of the software use. In many cases, the licenses can not be transferred to another organization. Sometimes a separate software license needs to be purchased for each individual unit that is being fielded, including operational sites, support sites, and test sites.

When developing the system, the Government needs to be sure that it will not cost an exorbitant amount of money for the licenses for the operational software. This is often accomplished by statements in the SOW requiring the developer to either (1) not include software in the system which will require licenses, (2) negotiate reasonable software licensing agreements, or (3) have the Government pay a one-time licensing fee for the software, so it can then be used in as many systems as the Government wants. Such SOW requirements are accompanied by requirements in Section L of the RFP requesting that the contractor state in his proposal how he is going to do this, and requirements in Section M stating that he will be evaluated on this.

For the support software, very few licenses will be required. In many cases, the Software Support Activity (SSA) will currently have the software license for that particular software; however, the contractor may not know this. This is avoided by including statements in the SOW requiring the contractor to provide information on what support software and other support equipment is necessary to support the operational software, what licenses are required, and how much the licenses will cost. In Section L of the RFP, there sould be statements requiring the contractor to include support software licensing information, and the related cost figures, in the proposal. In Section M of the RFP, there sould be statements to the effect that the offeror will be evaluated on this information.

2. Use Performance Based Requirements

Performance based requirements are stated in terms of the required performance or results for the system. These define the system's functional requirements, the environment(s) in which it will operate, interfaces to other systems, and interchangeability characteristics. A performance based specification states the requirements in terms of the required performance or results for the system, and provides criteria for verifying compliance. Performance based requirements do not specify “how to” go about achieving the required performance or results.

At the system level, requirements are defined and described in the System/Segment Specification (for DOD-STD-2167A and earlier systems) or the System/Subsystem Specification (for MIL-STD-498 systems). The performance based requirements that are in this document describe what the system will do. At this level, it is not specific as to which aspect or part of the system will do these requirements, whether hardware, software, or personnel. As the system is broken down into configuration items (CIs), each CI will have its own performance-based specification, stating what that CI will be doing. For Computer Software Configuration Items (CSCIs), these requirements are documented in the Software Requirements Specification (SRS) and Interface Requirements Specification (IRS). These documents provide the performance based requirements for the software; i.e., they state what the software in that CSCI is supposed to do, and not how it does it. Specific functions of the software, including operational and support functions, are specified, as well as timing requirements, processing requirements, user interface requirements, data base requirements, reliability requirements, safety requirements, security requirements, and anything else that is required of the software.

When developing the system, particularly a software intensive system, the system must include certain functions for protecting itself, the software within it, and the data that it processes from inappropriate disclosure, destruction, modification, and to prevent denial of service to the necessary users. This protection requires a balanced approach that includes system features as well as administrative, operational, physical, and personnel controls. Protection is commensurate with the classification level and category of the information, the threat, and the operational requirements associated with the environment of the operational system. To do this, appropriate security requirements need to be included in the system specification which states what system features will provide for or enhance the security. These requirements need to state what the system will do to protect itself from intentional or unintentional attack, the data, the software, or the functionality of the system. There also need to be statements in the SOW for the contractor to propose what administrative, operational, physical, and personnel controls are needed for the system when it becomes operational.

In terms of computer security requirements, DODD 5200.28 and SECNAVINST 5239.2A require that the system be built to at least the C2 Level of security as defined in DOD 5200.28-STD, "Department of Defense Trusted Computing System Evaluation Criteria", frequently called the "Orange Book". If the system is processing classified data, the C2 Level may not be sufficient; it may need to be developed to the B1 or B2 Level, or possibly even a higher level depending on the data that is being processed and the mode that it is operating in. These requirements include access restrictions, backup and recovery requirements, clearing and sanitization requirements, user identification, and accountability requirements, among others. These requirements must be specifically stated in the system specification.

The National Industrial Security Program Operating Manual (NISPOM), DOD 5220.22-M, of January 1995, is the replacement for the DOD Industrial Security Manual. The NISPOM is required to be invoked on every classified contract awarded by the Department Of Defense. This manual contains the industrial security requirements that the contractor must meet for his facilities, personnel, data, and other items involved in the development or support of the system. The NISPOM must be invoked on contracts which involve classified material; it does not have to be invoked on contracts which are entirely unclassified. Many of the requirements in the NISPOM are applicable to protecting sensitive unclassified data as well as classified data.

If special access requirements or clearances are necessary, or if the contractor will need to provide Secure Compartmented Information (SCI) spaces or facilities (SCIFs), then these requirements need to be specifically stated in the SOW. It is up to the contractor to obtain the special access billets required for the execution of the contract, obtain personnel and the appropriate clearances for these personnel, and obtain Government approval and accreditation of its facilities for use in the contract.

3. Use Integrated Product Teams

With acquisition reform initiatives, the Government is changing the way it acquires or develops software. The trend is moving away from having individual software engineers and programmers developing the software toward establishing teams where individuals from various disciplines get involved in the development of the software (as well as other components of the system). The trend is toward using Integrated Product Teams (IPTs). This is a team approach to systematically integrate and apply all disciplines throughout the system life cycle to produce an effective and efficient product or process that satisfies the stated requirements. In this type of arrangement, a few, or even one, individual involved with software development works on a team with engineers and individuals from other disciplines to develop an 'integrated product'.

Software IPTs are involved with ensuring that the software is developed properly and meets the requirements of the specifications, that it integrates correctly with the hardware, and also that it is safe, secure, and reliable within the context of the system that contains it. This IPT consists of the software members from the various other product IPTs; i.e., it crosses system or functional lines.

When IPTs are proposed, members will often be from the contractor (and subcontractors) as well as from the Government Program Management Office (PMO) and other Government activities involved in the acquisition. There may be several IPTs involved in the development of the system. When this is the case, the requirements for personnel, facilities, shared data bases, E-Mail, and communication for both the contractor and the Government need to be addressed. If DCMC representatives are included, they should not necessarily be the regular QAR representatives, but should be knowledgeable in software.

4. Emphasize Past Performance Information

Another initiative in acquisition reform is emphasizing past performance. The government’s ability to reliably predict future performance and to relax costly oversight procedures is directly tied to the offeror’s prior pertinent performance record.

When developing software for a system, it is preferred that the developer have experience in developing software for systems of this type, sometimes called the system 'domain'. This is a broad category of systems of the same or similar functions; common domains include command and control, communication, flight control, engine control, intelligence, radar, sonar, training, satellite, environmental, electronic warfare, explosive control, heavy equipment control, network, and other systems. Each of these domains may have sub-domains; for example, the communication systems domain has sub-domains of ELF, VLF, HF, UHF, SHF, ship-to-shore, air-to-air, satellite communications, submarine communications, secure communications, and other sub-domains in within the communications domain. Having experience in software development is often not sufficient; the RFP may state a preference for the developer to have

experience in software development of systems in the particular domain of the system that is being acquired. Also, the personnel that work for the contractor should have pertinent experience.

5. Manage Risk

A risk is the probability of an undesirable event occurring, and the impact of that event on the success of the project. It is the probability that, at any given point in the system life cycle, the predicted goals, whether cost, schedule, performance, or support, can not be achieved within the available resources.

Risk can never be totally removed from a software development effort. There are, however, techniques and methods that can be used to manage, mitigate or reduce risk, or to reduce the effects of an undesirable event if one does occur.

Risk can be controlled by effective communication, insight into product/process status, and taking prompt action on risky conditions. This includes the development of alternative courses of action ("Plan B").

There is also the possibility of making a conscious decision to accept the consequences of the undesirable event should that event occur. This is called risk assumption. Some amount of risk assumption always occurs in software development projects. This is usually done if the probability of the occurrence of the event is small, or the impact of the event occurring is minimal.

For events where the probability of the occurrence of the event is larger, or the impact is significant, the risks must be appropriately controlled. The risks that occur have to be identified as to what will happen, when they can possibly occur, how they can happen, why they can happen, what are the visible symptoms of them occurring or possibly occurring, and what will be the consequences if they do happen. There needs to be a way of determining if they are going to happen; risk indicators need to be identified and monitored throughout the development process, and even into the remainder of the life cycle of the system if necessary. If there are indications that something 'bad' might occur, actions will need to be taken to try to stop that event from occurring. There also need to be contingency plans to put in place should they occur anyway.

There are several areas of risk that frequently apply to software development projects. In many aspects, these risks apply to the program overall, not just to the software development effort. But it is in the software development area where these risks tend to be higher or tend to appear more frequently.

There are several methods that can be used in risk management in a software development effort.

- Improved “insight” through on-line real-time access to contractor management information.
- Practice better software development methods.
- Perform Independent Verification and Validation (IV&V).
- Designate a Risk Officer.
- Establish a joint contractor/government Computer Resources Working Group (CRWG)
- Develop and document risk management plans in a CRLCMP.
- Develop and use a Software Development Plan (SDP).

6. Streamline

Sometimes, when a standard is invoked in a contract, it is invoked in its entirety without tailoring to the specific acquisition. (Tailoring, in this case, means the deletion or non-inclusion of certain paragraphs of the standard that clearly do not apply to the system, and the modification of other paragraphs so that they apply to the requirements of the specific acquisition.) This has sometimes been the case with MIL-STD-498, "Software Development and Documentation", and its predecessor documents. MIL-STD-498 was written in a manner that allows for and encourages tailoring of the standard for application on specific contracts. For example, if the contract is only for requirements definition, only the paragraphs in Section 5 relating to requirements definition (and some paragraphs in the General Requirements section in Section 4) need to be invoked. MIL-STD-498, or whatever software development standard was invoked, needs to be tailored to this specific application.

With acquisition reform initiatives, tailoring is done by 'tailoring in', rather than 'tailoring out; in other words, tailoring by inclusion rather than tailoring by exclusion. This means that the specific sections or paragraphs of the document that is invoked need to be specifically cited. Previously, tailoring was done by citing the document and stating which paragraphs do not apply; now, the citations state only which paragraphs of the documents DO apply (and implying that the ones that are not stated do not apply to this contract). Therefore, whenever MIL-STD-498 is invoked, instead of stating 'in accordance with MIL-STD-498, except for Paragraphs 5.y and 5.z', the statements in the SOW should be similar to ".... in accordance with Paragraphs ____ of MIL-STD-498".

If another software development standard, such as the earlier DOD-STD-2167A, or the international standard ISO/IEC IS 12207, are invoked instead of MIL-STD-498, then the paragraphs of this document need to be appropriately tailored to the acquisition.

Increasingly, the Government will rely upon the contractor to submit as part of the proposal tailoring recommendations for the software development standard, and recommendations on how the standard would be applied to the contract, or portions of the contract. In order to do this, statements need to be included in Section L of the RFP requesting that offerors provide, (either as part of the draft SDP or separately) the tailoring recommendations for the software development standard as they apply to this specific acquisition. There also need to be statements in Section M which state that the offeror will be evaluated on these tailoring recommendations.

These tailoring recommendations are not limited to just the software development standard. Any standard that is invoked or given for guidance may be a candidate for tailoring recommendations. If this is what is desired, then there need to be statements in Section L of the RFP stating that the Government desires to have the contractor provide tailoring recommendations for certain standards, and list the standards. There also need to be statements in Section M which state that the contractor will be evaluated on these tailoring recommendations.

In many cases for CDRL items, not all of the information that is included in a DID is necessary for a specific acquisition. Any unnecessary information for that acquisition should be tailored out of the DID. Block 16 of the CDRL is where tailoring information for the DID is recorded. Any paragraphs of the DID which will not apply will be tailored out in the manner of "Delete Paragraphs 10.2.xxxx through 10.2.yyyy of DI-IPSC-zzzz". All of the paragraphs which need to be tailored out need to be listed here. Care must be taken to ensure that the entire contents of the DID are not tailored out; e.g., that it does not state "Delete Section 10.2 of the DID in its entirety".

Instances have also been noted of the creation of a 'pseudo-DID'; i.e., failure to use the DOD defined formal process of creating a DID or a one-time DID. This is done by stating in Block 16 something similar to "Delete Section 10.2.xxxx of the DID and replace by the following:". This is not allowed. DIDs can only be tailored down, i.e. paragraphs can only be deleted from the DIDs; paragraphs or requirements can not be added to the DIDs in the CDRL. If the PMO insists that they need a specific DID for one-time use on this one acquisition, then they need to create and obtain approval of a one-time DID; this is done through the Command Data Manager.

7. Use Commercial Practices, Products, And Processes

The system specification, Software Requirements Specifications, and Interface Requirements Specifications define the requirements for the system, and for the software within the system. In many cases, a Commercial Off-The-Shelf (COTS) software package can meet some or all of the requirements of the specification. The developer has to be sure that the functions that the COTS

software performs are the functions that are needed, and that it performs the functions correctly. The COTS package will usually not perform all of the functions required; if this is the case, either additional software will have to be developed to perform the remainder of the functions, or the PMO will have to make a decision as to whether these other functions really need to be performed. The cost of the COTS, including procurement of the COTS package, testing of the software, integrating the software with other software and with the rest of the system, and licenses for the COTS package all have to be taken into account when deciding to use COTS software, to modify previously-developed software, or to develop the software from scratch.

If software will be developed as part of this effort, then the contractor needs to have a defined software development process. Currently, this is done by invoking MIL-STD-498. Up to this point in time, there has been no commercial standard for software development processes; the only standards that existed were MIL-STDs. In many cases, commercial industrial organizations adopted these Military standards as their standard practice, making them in effect a 'best commercial practice', even though the standards had a DOD or MIL name on them. Starting in 1994, there have been efforts to develop non-DOD industry standards for software development processes. These efforts resulted in the still-ongoing work to develop IEEE P 1498, ISO/IEC 12207, and ANSI J STD 016, which will be the U.S. implementor of ISO/IEC 12207. When this standard comes out, this will be the one invoked on contracts, rather than MIL-STD-498. As with MIL-STD-498, this new standard is meant to be tailored to the specific acquisition. Only the appropriate requirements of the standard should be invoked.

Another practice is to include the use of a common programming language in writing the software. Until recently, DoD required software developers to obtain a waiver to use any programming language other than Ada 95 in new software. The current policy is that Ada must be considered when selecting a programming language, but the use of another language does not require a waiver. Ada was developed to provide a common programming language that would reduce development time and support costs during the life cycle of the system. If Ada 95 is selected as the programming language, then statements must be included in the SOW requiring the contractor to write the software in Ada 95, as defined by ANSI/ISO/IEC-8652:1995 "Information technology -- Programming languages -- Ada," and as such, is a commercial and international standard. In August, 1994, the DOD policy on the use of Ada was issued by USD(A&T) and ASD(C3I), which states that the Ada requirement does not conflict with the need to obtain a waiver for the use of MIL-STDs, and that the use of Ada must be cited in RFPs that involve software development as a required best practice. In April 1997, ASD(C3I) issued a memorandum “Use of the Ada Programming Language”, which states, in part, that DOD policy will be changed “to eliminate the mandatory requirement for use of the Ada programming language in favor of an engineering approach to selection of the language to be used”, and that “Ada should be one of the languages considered in this decision process; however, Ada waiver requests are no longer required when another language is selected.”

The Software Capability Maturity Model (CMM) was developed by the Software Engineering Institute (SEI) to enable organizations to determine how mature their software development

processes and methodologies are. The purpose of the CMM is to evaluate the process at certain contractors, sites, factories, divisions, or locations, to determine what level they are performing to and where improvements are needed. Sometimes, there have been references to the SEI CMM Model in RFPs, and requirements that the contractor's process be rated to a certain CMM level, usually Level 3, in order to be awarded contract. These requirements should not be included in the RFP, either as SOW requirements or as evaluation criteria in Section M. There should not be a requirement that the contractor, in order to get the contract, be performing at a certain level. What needs to be done is that the contractor should be asked to provide information as to whether a CMM Assessment was done, and if so, what the results were. This information should be asked for in Section L of the RFP, and included as an evaluation factor in Section M. These results are then included as part of the evaluation during source selection.

8. Use On-Line Electronic Media

When a system is being developed for the Government, a lot of data will be generated during the development process. In many cases, the Government reviews this data to be able to review and approve the development plans (such as the Software Development Plan), to get reports on the progress of development, to review the development of the specifications and the design, to review user procedures, or other data that it determines it needs to allow for insight of the development effort, to allow for the support of the system, or to allow for the reprocurement of the system from another contractor. In the past, in order for the Government to get the data, there had to be a Contract Data Requirements List (CDRL), which would list each individual item of data that is to be delivered to the Government.

In many instances, the Government needs to get a copy of a document, or certain data, from the contractor to gain insight as to what the contractor is doing and how well he is performing. In the past, this information was obtained by asking for a hard copy of certain documents, such as Software Development Files (SDFs), in the CDRLs. With Electronic Document Interchange (EDI) capabilities, reviewing this data can be done easily through computers and networks. The Government is able to electronically review a document that is resident on a contractor's computer and obtain a copy or provide comments on the document. This information may include status reports, cost reports, metrics, and other management data as well as design information and test information. These items have been included in the CDRLs in the past so the Government would get a copy of this information. If the Government has on-line access to this information, there is no need to have it officially delivered to the Government. This saves time, money and frustration.

This does not mean that the contractor will allow totally unrestricted access to the system; appropriate security requirements still have to be observed. In most cases, the access will be 'read only' access to other than the personnel actually developing or modifying that document. In many cases, even the 'read only' access may be restricted only to those allowed to view the document, or those approved by the Program Manager to review that specific item. These

statements in the SOW should require the contractor to allow Government personnel to access the document, make copies of the document, and provide comments on the document, but should not allow the Government personnel to modify the document.

If this is what is desired, then there need to be requirements included in the SOW establishing a preference for the contractor to provide the Government access to its data on its computers, the data and information that the Government prefers to access should be specifically stated. If this is done, then there is no need to include these items in the CDRL list, because the data will not be officially 'delivered' to the Government. The Government will have access to the data during the course of development, so it does not need an 'officially delivered' document for this information. The CDRL list needs to be examined for these types of documents. If the information that is normally in these documents can be more easily obtained through on-line access, there is no need to have a CDRL item for this information.

Even though much of the data can be delivered to or viewed by the government electronically, there will be cases where a CDRL is required for certain documents. This is particularly the case where data or documents, such as specifications, must be officially delivered to and "accepted" by the Government. The term "Acceptance" has an official definition in the FAR: "An action by an authorized representative of the Government by which the Government assumes ownership of the item as partial or complete performance of a contract.". The key term here is "assumes ownership". If the title to the item must be officially transferred to the Government, then a DD 250 is required to be signed by the authorized Government representative after the delivery of the item and after the item has been inspected and found to meet the requirements. A DD 250, Certificate of Inspection and Acceptance, is the legal document whereby the title of the item, specification, drawing, or other data, is transferred from the developer to the Government. In these instances, a CDRL item is still required for the document, because the item must be officially delivered to and accepted by the Government. In these instances, a "DD", "SD", "DS", or "SS" must be given in Block 7 of the CDRL; the first letter indicates the inspection location, source (S) or destination (D), and the second letter indicates the acceptance location, source (S) or destination (D). Generally, the practice is specifications, drawings, and related data are 'accepted'; plans, designs, and some reports are 'approved'.

In the cases where approval is required, such as for plans or designs, an "A" must be included in Block 8 of the CDRLs. Without the "A" in Block 8, the Government does not have approval authority of the document. The contractor can deliver the document and the Government has no contractual right to go back to the contractor to tell him to make changes, or even to say whether it is good or bad. When reviewing an RFP, it must be ensured that the appropriate documents that require approval by the Government have an "A" in Block 8 of their CDRL listing.

For software, the only documents that fit this description are the System/Segment Specification (DOD-STD-2167/2167A), System/Subsystem Specification (MIL-STD-498), Software Requirements Specification, Interface Requirements Specification, and Software Product Specification. The number of documents that are to be acquired via the CDRL needs to be

minimized to the smallest extent practical, consistent with FAR Regulations and other policy and guidance.

The CDRL list needs to be reviewed to ensure that all documents on the list actually need to be officially delivered. If the document can be obtained in another manner, such as on-line through the contractor's data processing system, there is probably no need to have this document in the CDRL list. If the document contains only management or tracking data, this data should be available through the on-line system. In this case, there is no need to officially have delivery of the information (i.e., the document containing this information). The SOW needs to be reviewed to ensure that there are appropriate statement included in it that require the contractor to allow Government access to the data in its systems. For those CDRL items that are to be acquired, the SOW needs to contain statements requiring the contractor to perform the work which is documented by those documents. The SOW needs to have the contractor 'do the work', not 'develop the document'.

With modern technology, in most cases the document that will be delivered is in electronic form on a machine that the Government can access. This is especially true of software programs and data bases, as well as their related documentation. In the past, these documents would be printed out and then delivered in "hard copy" format to the Government. The software programs would be compiled, and a compilation listing of the program (usually on 11" x 17" computer paper) would be delivered to the Government. With the current state of the art of interconnected and networked systems, it is possible to send documents (or more appropriately, information) electronically rather than send paper "hard copy" documents. As such, when the document is finished, it can be 'delivered' to the Government electronically, rather than in hard copy. The preferred medium for delivery of the item is electronic.

This speeds up the delivery time of the document to the Government. It reduces the number of copies that need to be produced. It reduces shipping and mailing costs for large and/or numerous documents. In the past, the contractor would send the document by mail (sometimes even certified, registered, or 'return receipt requested') to the Government with a cover letter stating something similar to "In accordance with contract number xxxx, CDRL AYYY is hereby submitted to the Government."

With current technology, it can be delivered to the Government in several alternative ways. The method of delivery must be stated in Block 16 of each CDRL item. This information needs to specifically state how and in what format the document will be delivered. If it is to be delivered on a floppy disk, then the physical size, storage size, computer system compatibility (e.g. PC with MS-DOS, Windows, Macintosh, UNIX, etc.), and format requirements for the disk need to be stated. If it is to be delivered by electronic mail, then the requirements for compatibility with specific electronic mail and/or encryption software need to be stated. If it is to be delivered by hard copy in addition, this also needs to be stated. The individuals or organizations that the contractor sends the document to must be listed in Block 14 of the CDRL. Another way is to have the contractor state which file the document resides in on his computer, and have the

Government personnel who need to see it obtain a copy. In this case, the contractor can send an E-MAIL message to the appropriate addressees stating something similar to "In accordance with contract number xxxx, the requested information is now available to the Government in electronic format. It can be found on our machine in directory and file \widget\cdrl\ayyy.doc. This document is in _____ format." The individuals or organizations that are listed in Block 14 of the CDRL must be able to have access to this file.

There are cases where the Government needs the information, but does not care about the format or layout of the information. Any format the contractor chooses to submit it in is acceptable, as long as it contains all the necessary information. If the format does not matter, then the phrase "Contractor format acceptable" is usually included as the first statement in Block 16. Care must be taken when using this. There are some documents, such as specifications and technical manuals, where there needs to be a specific format. In these instances, these documents must adhere to this format, and any contractor defined format is not allowed.

With the availability of the Internet, electronic bulletin boards, and other easily accessible ways of obtaining information, the temptation is great to put system documents and other information on an easily accessed website. This should be done only if the website meets the security requirements of CNO message 212001Z Jul 95, Guidelines for Naval use of the Internet, and the references cited therein (http://www.chinfo.navy.mil/navpalib/internet/navyinet.txt).

9. Use Unobtrusive Testing Techniques

Testing of the system must be done to ensure that the system complies with its specified requirements, and that it is operationally effective and suitable. As part of the system, the software must also be tested. Software testing differs from hardware testing, just as software differs from hardware. Whereas hardware is the physical component of the system, the software is the logical sequential flow of operations that are to be done. Hardware testing tests the physical make-up and construction of the system; software testing tests the logical operation of the instructions. The testing of software (as software, prior to integration with the system) is done on many levels, from unit level through CSCI level. Experience has shown that many software problems are discovered in the lower levels of testing. The levels of testing that are done on the software, and the extent that the testing is done on each of these levels, is system dependent. Testing does not need to be unnecessarily obtrusive. A risk avoidance assessment should be done to determine the appropriate levels and redundancy of testing required for the software. The development contractor may also have his own standard practices, processes, and procedures for conducting software testing. Rather than telling the contractor how to go about testing the software in the system, the RFP should request that the contractor propose what it is he intends to do, and describe the practices, processes, and procedures that he will use for conducting software testing. Testing does not need to be unnecessarily obtrusive. A risk assessment should be done to determine the appropriate levels and redundancy of testing required for the software.

Another issue related to software testing is the potential modification of the COTS or NDI software. In many cases, the previously developed software will almost meet the requirements of the new system. Some function(s) may need to be added, or some small changes may need to be made in order to make the software work in the new system. If this software needs to be modified, then the modifications will, in most cases, be treated as a new software development effort. The new or modified code will have to be designed, developed, implemented, tested, and integrated just like any other newly developed software. This is part of the overall software development effort. In order to do this, there need to be statements in the SOW regarding modification of COTS/NDI software. If modifications need to be done, then it will be treated as new development, the defined software development process (e.g. MIL-STD-498) will apply, and these changes will have to be appropriately tested. There also need to be requirements in Section L requiring that the contractor provide information regarding his plans to modify any COTS/NDI software needing modification in the draft SDP, and statements in Section M stating that this information will be evaluated as part of the evaluation process.

10. Use Best Value Source Selection Techniques

"Best Value" is a source selection process which allows for a "trade-off" between cost and non-cost factors (e.g. technical approach, past performance). Regarding software, this means the greatest overall value to the government, over the life cycle of the system, may not be the offer with the lowest proposed initial acquisition cost. For example, an offeror may have a significantly better Software Development Plan for developing/acquiring necessary software, have a better pertinent past performance record, or higher Software Capability Maturity (CMM) level in accordance with Software Engineering Institute (SEI) assessments. Such factors may offset initial proposal price differences if carefully documented in accordance with RFP section L and M provisions.